Form: PTO/SB/17 (Modified)

# REPLY/AMENDMENT FEE TRANSMITTAL

| | |
|---|---|
| Attorney Docket No. | 95-427 |
| Application Number | 09/604,880 |
| Filing Date | June 28, 2000 |
| First Named Inventor | WHEELER |
| Group Art Unit | 2122 |
| Examiner Name | GROSS, Kenneth |

| | |
|---|---|
| AMOUNT ENCLOSED | $ 0 |

## FEE CALCULATION (fees effective 10/01/2001)

| CLAIMS AS AMENDED | Claims Remaining After Amendment | Highest Number Previously Paid For | Number Extra | Rate | Calculations |
|---|---|---|---|---|---|
| TOTAL CLAIMS | 26 | 26 | 0 (3) | X $18.00 = | $0 |
| INDEPENDENT CLAIMS | 4 | 4 | 0 | X $84.00 = | $0 |

Since an Official Action set an original due date of____, petition is hereby made for an extension to cover the date this reply is filed for which the requisite fee is enclosed (1 month ($110); 2 months ($400); 3 months ($920); 4 months ($1,440); 5 months ($1,960)):



| | |
|---|---|
| If Statutory Disclaimer under Rule 20(d) is enclosed, add fee ($110) | |
| Total of above Calculations = | $0 |
| Reduction by 50% for filing by small entity (37 CFR 1.9, 1.27 & 1.28) | - |
| **TOTAL FEES DUE** = | $0 |

(1) If entry (1) is less than entry (2), entry (3) is "0".
(2) If entry (2) is less than 20, change entry (2) to "20".
(4) If entry (4) is less than entry (5), entry (6) is "0".
(5) If entry (5) is less than 3, change entry (5) to "3".

## METHOD OF PAYMENT

[ ] Check enclosed as payment.

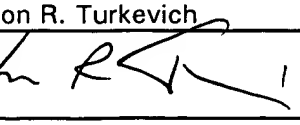
[ ] Charge "TOTAL FEES DUE" to the Deposit Account No., below.

## AUTHORIZATION

[X] If the above-noted "AMOUNT ENCLOSED" is not correct, the Commissioner is hereby authorized to credit any overpayment or charge any additional fees under 37 CFR 1.16 or 1.17 necessary to maintain pendency of the present application to:

| | |
|---|---|
| Deposit Account No.: | 50-1130 |
| OrderNo.: (Client/Matter) | 95-427 |

**SUBMITTED BY: LEON R. TURKEVICH, ESQ.**

| | | | |
|---|---|---|---|
| Typed Name | Leon R. Turkevich | Reg. No. | 34,035 |
| Signature |  | Date | June 18, 2004 |

Docket No.: 95-427 PATENT

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

| | | |
|---|---|---|
| In re Application of | : | |
| WHEELER, et al. | : | |
| Serial No.: 09/604,880 | : | Group Art Unit: 2122 |
| Filed: June 28, 2000 | : | Examiner: GROSS, Kenneth A. |

For: GENERIC COMMAND INTERFACE FOR MULTIPLE EXECUTABLE ROUTINES

**RESPONSE**



Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the Nonfinal Official Action mailed March 18, 2004, the following remarks are submitted.

Reconsideration and allowance of the above-referenced application are respectfully requested. Claims 1-26 are unchanged and remain pending in the application.

Claims 1-26 stand rejected under 35 USC §112, first paragraph. This rejection is respectfully traversed. The rejection specifically refers only to claim 1, and summarily incorporates the rejection to claims 2-26. In particular, the rejection asserts that "claim 1 recites 'receiving a generic command from the user' where the generic command is validated based on 'a prescribed generic command format'" and argues that "It is unclear how a generic command can have a prescribed format."

However, claims 1, 10, and 14 specify a command parse tree (claim 23 is addressed below). The command parse tree specifies valid generic commands relative to a prescribed generic command format. Further, the command parse tree has elements, each specifying at least one generic command component and a corresponding at least one command action value. As described in the specification at page 4, lines 16-22:

> the parser 14 and the translators 16 provide a generic command syntax that integrates the functionality of the different tools 18 and that automatically selects the appropriate command for the best tool for executing a given generic command. As illustrated in Part A. of the attached appendix, the new syntax provides a generic instruction set that provides an abstraction of the tool-specific command formats and syntax, enabling a user to issue command based on the relative functions, as opposed to the specific syntax for a corresponding tool 18.

Hence, the specification describes that the parser 14 and the translators 16 provide a generic command syntax (e.g., a generic instruction set illustrated in Part A of the appendix) that provides an abstraction of the tool-specific command formats and syntax. Hence, to respond to the Examiner's query, a command with a "prescribed command format" is not inherently or even necessarily a specific command, but rather can still be an abstraction of the tool-specific command format and syntax.

Part A of the Appendix specifies numerous examples where a prescribed functional item ("Functional Item") having a tool-specific command format and syntax ("Old Command Line/Syntax") is replaced with a "New Syntax" that is an abstraction of the tool-specific command format and syntax: as an example, note that the "Old Command Line/Syntax" commands ("BASEview/BASEview -G/BASEview -h then press g" and "APPview/APPview -g/APPview -i then press g") for the respective Functional Items "Watch BASE Global

Client/Server Information" and "Watch APP Global Information" are mapped to the same New Syntax command "watch acb globals".

Hence, multiple functions can be mapped to new syntax commands. Also note that the fact that execution of a generic command "watch acb globals" may initiate execution of multiple functional items may be deemed by one skilled in the art an acceptable by-product of using the generic commands.

Regarding the issue of validating the generic command "based on a command parse tree that specifies valid generic commands relative to a prescribed generic command format", attention is directed to Figure 2, which discloses that the parser 14 includes a command word translation table 20 and a command parse tree 22. As described at page 4, lines 27-29, "the command word translation table 20 includes all the command words 26 that are valid according to the generic syntax, illustrated for example in Part B of the attached appendix."

Part B of the attached appendix illustrates examples of generic commands, where each generic command has a corresponding identified usage (i.e., syntax) with prescribed acceptable parameters:

Watch <Object> [Screen]
Get <Variable>
Set <Variable> <Value>
Start <Agent>
(etc.)

Hence, the command word translation table 20 identifies the valid command words and acceptable parameters. Note, however, that even though the command word translation table 20 may not necessarily identify the syntax (e.g., ordered sequence of parameters relative to command words), the parser 14 includes a command parse tree having a structure

(interconnected tree elements 24) that implements the syntax illustrated in Part B of the Appendix:

> The parser 14 is configured for validating a received generic command by comparing each input command word to the command parse tree 22 to determine for the received generic command a tree element 24 identified as a best match. Each tree element 24 includes at least one token-command key pair 30 that specifies a token (T) 28 and a corresponding command key (CK) 32, enabling the parser 14 to identify the appropriate prescribed command based on the command key specified for the matching token. In particular, the parser 14 recursively traverses the command parse tree 22 for each command word to identify the best match for the generic command. If only a portion of the generic command is identified as valid (e.g., only the first three command words are valid), the parser 14 selects the command key 32 for the matching token 28 from the last valid tree element 24.
>
> (Page 5, lines 1-10).

Finally, Figure 3 and the accompanying text at page 5, line 11 to page 6, line 19 provide specific examples for validating the supplied commands "watch tcp connections" (deemed valid) and "get udp connection info" (deemed partially valid, partially invalid). In each case a match is identified (e.g., a tree element 24), and the corresponding action value (e.g., command key 30) is issued for an identified one of the protocol-specific translators 16, for protocol-specific translation for a corresponding management program 18.

One having ordinary skill in the art, upon reviewing the specifciation, would recognize that validation of the generic command can be readily implemented (without undue experimentation) by using a command parse tree having a structure that impelements a prescribed generic command format (e.g., as illustrated in Part B of the Appendix).

Claim 23 specifies means for validating a generic command received from a user. This means for validating reads on the disclosed structure, namely the disclosed parser 14 of Figure 2

which includes the command word translation table 20 and the command parse tree 22; hence, the description above with respect to claims 1, 10, and 14 is incorporated herein by reference.

For these and other reasons, claims 1-26 comply with the requirements of 35 USC §112, first paragraph. Hence, this rejection should be withdrawn.

Claims 1, 10, 14, and 23 stand rejected under 35 USC 103(a) in view of U.S. Patent No. 6,138,098 to Shiber et al. and U.S. Patent No. 6,516,356 to Belknap et al. This rejection is respectfully traversed.

As admitted in the Official Action, Shieber does not teach that the command is a generic command. Moreover, as described in the Response filed October 3, 2003 (the comments of which are incorporated in their entirety herein by reference), Shieber does not disclose or suggest "validating the generic command based on a command parse tree that specifies valid generic commands relative to a prescribed command format" or "issuing a prescribed command of a selected one of the management programs according to the corresponding command format, based on the identified one element," as claimed. Rather, Scheber et al. discloses a method allowing a user to control a single computer application at a time using spoken commands. The spoken commands are converted by a speech recognition application 37 into at least one candidate word phrase; the candidate word phrase is parsed with a Context Free Grammar parser into a parse tree. (See pages 2-4 of the Response filed October 3, 2003).

The Official Action cites Belknap for the proposition that it teaches "inputting a generic command into a command parser, which then applies a specific command to a specific media device." This naked description of Belknap, however, is insufficient to render the claimed invention obvious.

Belknap et al. discloses a media manager 5 (see Fig. 1) having a high-level command processor 10 configured for decoding a high-level generic command, received from a requesting application 8, to identify whether a command corresponds to a particular type of media server which the media manager supports. If the command processor 10 determines that the command corresponds to a media device 25 supported by the media manager 5, the processor 10 passes the request to a device-specific code-mapping module 15 which handles the high-level commands destined for that media device 25 (see Fig. 1 and col. 2, line 52 to col. 3, line 8).

In particular, Figure 2 illustrates in step 50 that the high-level command is received by the media manager 5 from the requesting application 8, along with a target media device: the server type (e.g., video or audio based file) is determined by the high-level command processor in step 55 based on an identifier in the high-level command (see col. 5, lines 41-48 and col. 7, lines 1-4).

Hence, Belknap teaches that a requesting application can send a high-level request, where the high-level command processor determines the server type (e.g., video or audio based file) based on an identifier in the high-level command.

The Official Action fails to provide a prima facie case of obviousness because there is no evidence that one having ordinary skill in the art would have been motivated to modify Shieber et al. in order to include the teachings of Belknap et al.

In particular, the Examimer asserts the supposed motivation to be that the teachings of Belknap "allows a level of abstraction for specifying commands for a plurality of programs with different command formats." However, this alleged motivation is improper because it is the

exact advantage attributed to the claimed features in the subject application at page 4, lines 19-22:

> [T]he new syntax provides a generic instruction set that provides an abstraction of the tool-specific command formats and syntax, enabling a user to issue command based on the relative functions, as opposed to the specific syntax for a corresponding tool 18.

(see also page 1, line 25 to page 2, line 2; page 2, lines 10-13.

There is no evidence that this advantage would have been apparent or known to one skilled in the art at the time the invention was made. Hence, use of Applicant's own specification as the supposed motivation for combining references is improper. "It is impermissible to use the claimed invention as an instruction manual or 'template' to piece together the teachings of the prior art so that the claimed invention is rendered obvious." In re Fritch, 23 USPQ2d 1780, 1784 (Fed. Cir. 1992).

Moreover, the supposed motivation fails to identify why one skilled in the art would have been motivated to modify the primary reference (Shieber et al.) to include the teachings of Belknap et al. "Teachings of references can be combined only if there is some suggestion or incentive to do so." In re Fine, 5 USPQ2d 1596,1600 (Fed. Cir. 1988) (quoting ACS Hosp. Sys. v. Montefiore Hosp., 221 USPQ 929, 933 (Fed. Cir. 1984)) (emphasis in original).

In fact, one skilled in the art would have avoided combining the teachings of the two references as mutually inconsistent: Shieber et al. is directed to allowing a user to control a computer application with spoken commands, wherein a speech recognition application processes the spoken commands into candidate word phrases, and where at least one candidate word phrase is parsed by a context free grammar parser into a parse tree, in order to overcome

the word ambiguity and speech misinterpretation that occurs in speech recognition (see col. 5, line 52 to col. 6, line 12).

Belknap, however, is directed to a multimedia invironment where a requesting software application 8 sends a high level command which specifies the media type. As apparent from the foregoing, the respective teachings of Shieber et al. and Belknap et al. are inconsistent and incompatible: Shieber receives unknown spoken words and uses a parse tree to determine a context for spoken text from a user; Belknap receives a software command from an executable application, where the software command specifies an identifier identifying a media type.

Hence, the Official Action fails to provide any evidence as to why one skilled in the art would even want to combine the teachings of the two references, especially in view of the mutually exclusive inputs (spoken text by user vs. command for specified media by an executable software applicaton) and the mutually exclusive results (determine the spoken words vs. output a media-specific command to a media-specific media server). "The mere fact that the prior art may be modified in the manner suggested by the Examiner does not make the modification obvious unless the prior art suggested the desirability of the modification." In re Fritch, 23 USPQ2d 1780, 1783-84 (Fed. Cir. 1992).

Moreover, assuming one skilled in the art would have been motivated to combine the two references, the resulting hypothetical combination still would neither disclose nor suggest the claimed features of validating the generic command based on a command parse tree that specifies valid generic commands relative to a prescribed generic command format, as claimed. There is no disclosure or suggestion in Shieber et al. or Belknap et al., singly or combined, of a command parse tree that specifies valid generic commands. As described previously, the parse tree in

Shieber et al. is continually rewritten -- hence, the parse tree 54 in Shieber et al. is incapable of specifying "valid generic commands relative to a prescribed generic format" because the parse tree 54 is continally rewritten using a series of phrases 80.

An evaluation of obviousness must be undertaken from the perspective of one of ordinary skill in the art addressing the same problems addressed by the applicant in arriving at the claimed invention. Bausch & Lomb, Inc. v. Barnes-Hind/Hydrocurve, 23 USPQ 416, 420 (Fed. Cir. 1986), cert. denied, 484 US 823 (1987). Thus, the claimed structures and methods cannot be divorced from the problems addressed by the inventor and the benefits resulting from the claimed invention. In re Newell, 13 USPQ2d 1248, 1250 (Fed. Cir. 1989).

Neither of the references, singly or in combination, address the problem of executing a plurality of management programs according to respective command formats, where a generic command can be converted to a prescribed command for a selected management program: Shieber et al. is concerned with word recognition in a speech recognition system, and Belknap is concerned with accomodating interactions with multiple media servers by a requesting application that is able to specify the specific media type being requested.

For these and other reasons, the §103 rejection of claims 1, 10, 14, and 23 should be withdrawn.

Claims 2-9, 11-13, 15-22, and 24-26 stand rejected under §103 in view of Shieber et al., Belknap et al., and U.S. Patent No. 6,397,283 to Hancock et al. It is believed these dependent claims are allowable in view of their dependency from their respective independent claims.

Further, the Official Action fails to provide evidence that one of ordinary skill in the art would have been motivated to modify Shieber and Belknap et al. to include the teachings of

Hancock. In fact, Hancock is not a related technology to Shieber, since Shieber is directed to speech recognition, and Hancock is directed to interfacing with logical device drivers with a computer.

Regardless, the hypothetical combination still would neither disclose nor suggest the claimed features. For example, Hancock neither disclsoes nor suggests the feature in claims 2, 11, 15, 24, of comparing each input command word to a command word translation table, configured for storing for each prescribed command word a corresponding token, for identification of a matching token, and determining a presence of a matching token within the command parse tree. Rather, Hancock teaches that the command table of Figure 6 includes "m" command entries, with each command entry having "n" tokens (col. 8, lines 51-54). Hence, Hancock neither discloses nor suggests a corresponding token for each prescribed command word, as claimed.

More fundamentally, the hypothetical combination neither solves nor addresses the problems contemplated by the inventor, namely enabling a simple command language to be utilized for control of multiple real time monitoring programs having respective command formats, or providing an arrangement that integrates multiple RTM programs and command and control functionality for a user, without the necessity of learning the respective command formats and syntax. An evaluation of obviousness must be undertaken from the perspective of one of ordinary skill in the art addressing the same problems addressed by the applicant in arriving at the claimed invention. Bausch & Lomb, Inc. v. Barnes-Hind/Hydrocurve, 23 USPQ 416, 420 (Fed. Cir. 1986), cert. denied, 484 US 823 (1987). Thus, the claimed structures and methods

cannot be divorced from the problems addressed by the inventor and the benefits resulting from the claimed invention. In re Newell, 13 USPQ2d 1248, 1250 (Fed. Cir. 1989).

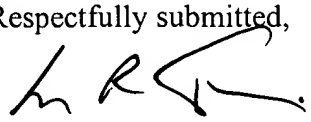
The Official Action fails to provide any recognition of the problems associated by the inventors, but merely asserts the naked argument that "a table allows for fast lookup and parsing of a command". Hence, the Official Action fails to establish a prima facie case of obviousness, because the hypothetical combination of Shieber et al., Belknap et al. and Hancock, would be inoperative in view of the mutually inconsistent technologies and required inputs and outputs as described above. Regardless, the hypothetical combination would neither disclose nor suggest the claimed system for executing a plurality of management programs according to respective command formats based receiving a generic command from a user.

For these and other reasons, the §103 rejection should be withdrawn.

In view of the above, it is believed this application is in condition for allowance, and such a Notice is respectfully solicited.

To the extent necessary, Applicant petitions for an extension of time under 37 C.F.R. 1.136. Please charge any shortage in fees due in connection with the filing of this paper, including any missing or insufficient fees under 37 C.F.R. 1.17(a), to Deposit Account No. 50-1130, under Order No. 95-427, and please credit any excess fees to such deposit account.

Respectfully submitted,



Leon R. Turkevich
Registration No. 34,035

Customer No. 23164
**Date: June 18, 2004**